Convex Programming by Gradient Descent

Zhi Li

May 23, 2018

Abstract

This article focuses on proofs.At first,we consider GD algorithm in the unconstrained situation, where f is a convex and β-smooth function on R^n. Then we will analyze SGD algorithm when it is invoked to minimize a strongly convex objective function, where it is possible to establish a global rate of convergence to the optimal objective value.

1 Introduction

Many situations arise in machine learning where we would like to optimize the value of some function. That is, given a function $f : R^n \to R$, we want to find $x \in R^n$ that minimizes (or maximizes) $f(x)$. It turns out that, in the general case,finding the global optimum of a function can be a very difficult task. However, for a special class of optimization problems known as convex optimization problems, we can efficiently find the global solution in many cases. The central object of this article is convex programming by gradient descent.

2 Basic Concepts of Convex Programming [1]

2.1 Convex Sets

Denition 2.1 set C is convex if, for any $x, y \in C$ and $\theta \in R$ with $0 \leq \theta \leq 1$,

$$\theta x + (1 - \theta x)y \in C \tag{1}$$

Intuitively, this means that if we take any two elements in C, and draw a line segment between these two elements, then every point on that line segment also belongs to C.

2.2 Convex Fuction

Denition 2.2 function $f : R^n \to R$ is convex if its domain (denoted $D(f)$) is a convex set, and if, for all $x, y \in D(f)$ and $\theta \in R, 0 \leq \theta \leq 1$,

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y) \tag{2}$$

Intuitively, the way to think about this denition is that if we pick any two points on the graph of a convex function and draw a straight line between then, then the portion of the function between these two points will lie below this straight line.We say a function is strictly convex if Denition 2.1 holds with strict inequality for $x \neq y$ and $0 < \theta < 1$.

2.3 First Order Condition for Convexity

Suppose a function $f : R^n \to R$ is differentiable. Then f is convex if and only if $D(f)$ is a convex set and for all $x, y \in D(f)$,

$$f(y) \geq f(x) + \nabla_x f(x)^\top (y - x) \tag{3}$$

The function $f(x)+\nabla_x f(x)^\top (y-x)$ is called the first-order approximation to the function f at the point x. Intuitively, this can be thought of as approximating f with its tangent line at the point x. The first order condition for convexity says that f is convex if and only if the tangent line is a global underestimator of the function f. In other words, if we take our function and draw a tangent line at any point, then every point on this line will lie below the corresponding point on f. Similar to the definition of convexity, f will be strictly convex if this holds with strict inequality, concave if the inequality is reversed, and strictly concave if the reverse inequality is strict.

2.4 Second Order Condition for Convexity

Suppose a function $f : R^n \to R$ is twice dierentiable. Then f is convex if and only if $D(f)$ is a convex set and its Hessian is positive semidefinite. i.e., for any $x \in D(f)$

$$\nabla_x f(x) \geq 0 \tag{4}$$

3 Gradient Descent for Smooth Functions [2]

We say that a continuously differentiable function f is β-smooth if the gradient ∇f is β-Lipschitz, that is

$$\|\nabla f(x) - \nabla f(y)\| \leq \beta \|x - y\| \tag{5}$$

In order to avoid technicalities we consider the unconstrained situation, where f is a convex and β -smooth function on R^n.
Theorem 3.1 Let f be convex and β-smooth on R^n. Then gradient descent with $\eta = \frac{1}{\beta}$ satisfies

$$|f(x_t) - f(x*)| \leq \frac{2\beta \|x - x*\|^2}{t-1} \tag{6}$$

Before embarking on the proof we state a few properties of smooth convex functions.

Lemma 3.1 Let f be a β-smooth function on R^n . Then for any $x, y \in R^n$, one has

$$\big|f(x) - f(y) - \nabla f(y)^\top (x-y)\big| \leq \frac{\beta}{2}\|x-y\|^2 \tag{7}$$

Proof. We represent $f(x) - f(y)$ as an integral, apply Cauchy-Schwarz and then β-smoothness:

$$|f(x) - f(y) - \nabla f(y)^\top (x-y)| = |\int_0^1 \nabla f(y + t(x-y))^\top (x-y)\, dt - \nabla f(y)^\top (x-y)|$$

$$\leq \textstyle\int_0^1 \|\nabla f(y + t(x-y)) - \nabla f(y)\| \cdot \|x-y\|\, dt$$

$$\leq \textstyle\int_0^1 \beta t \|x-y\|^2\, dt$$

$$= \frac{\beta}{2}\|x-y\|^2 \tag{8}$$

In particular this lemma shows that if f is convex and β-smooth, then for any $x, y \in R^n$, one has

$$0 \leq f(x) - f(y) - \nabla f(y)^\top (x-y) \leq \frac{\beta}{2}\|x-y\|^2 \tag{9}$$

This gives in particular the following important inequality to evaluate the improvement in one step of gradient descent:

$$f\Big(x - \frac{1}{\beta}\nabla f(x)\Big) - f(x) \leq -\frac{1}{2\beta}\|\nabla f(x)\|^2 \tag{10}$$

Lemma 3.2 . Let f be such that (9) holds true. Then for any $x, y \in R^n$, one has

$$f(x) - f(y) \leq \nabla f(x)^\top (x-y) - \frac{1}{2\beta}\|\nabla f(x) - \nabla f(y)\|^2 \tag{11}$$

Proof. Let $z = y - \frac{1}{\beta}(\nabla f(y) - \nabla f(x))$. Then one has

$$\begin{aligned} f(x) - f(y) &= f(x) - f(z) + f(z) - f(y) \\ &\leq \nabla f(x)^\top (x-z) + \nabla f(y)^\top (z-y) + \frac{\beta}{2}\|z-y\|^2 \\ &= \nabla f(x)^\top (x-y) + (\nabla f(x) - \nabla f(y))^\top (y-z) \\ &+ \frac{1}{2\beta}\|\nabla f(x) - \nabla f(y)\|^2 \\ &= \nabla f(x)^\top (x-y) - \frac{1}{2\beta}\|\nabla f(x) - \nabla f(y)\|^2 \end{aligned} \tag{12}$$

We can now prove Theorem 3.1
Proof. Using (10) and the denition of the method one has

$$f(x_{s+1}) - f(x) \leq -\frac{1}{2\beta}\|\nabla f(x_s)\|^2 \tag{13}$$

In particular, denoting $\delta_s = f(x_s) - f(x^*)$ this show:

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\beta}\|\nabla f(x_s)\|^2 \tag{14}$$

One also has by convexity

$$\delta_s \leq \nabla f(x_s)^\top (x_s - x^*) \leq \|x_s - x^*\| \cdot \|\nabla f(x_s)\| \tag{15}$$

We will prove that $\|x_s - x^*\|$ is decreasing with s, which with the two above displays will imply

$$\delta_{s+1} \leq \delta_s - \frac{1}{2\beta\|x_1 - x^*\|^2}\delta_s^2 \tag{16}$$

Let us see how to use this last inequality to conclude the proof. Let $w = \frac{1}{2\beta\|x_1 - x^*\|^2}$,then

$$w\delta_s^2 + \delta_{s+1} \leq \delta_s \Leftrightarrow w\frac{\delta_s}{\delta_{s+1}} + \frac{1}{\delta_s} \leq \frac{1}{\delta_{s+1}} \Rightarrow \frac{1}{\delta_{s+1}} - \frac{1}{\delta_s} \geq w \Rightarrow \frac{1}{\delta_t} \geq w(t-1) \tag{17}$$

Thus it only remains to show that $\|x_s - x*\|$ is decreasing with s.Using (10) one immediately gets

$$(\nabla f(x) - \nabla f(y))^\top (x - y) \geq \frac{1}{\beta}\|\nabla f(x) - \nabla f(y)\|^2 \tag{18}$$

We use this as follows (together with $\nabla f(x*) = 0$)

$$\begin{aligned}
\|x_{s+1} - x*\|^2 &= \|x_s - \frac{1}{\beta}\nabla f(x) - x*\|^2 \\
&= \|x_s - x*\|^2 - \frac{2}{\beta}\nabla f(x)^\top (x_s - x*) + \frac{1}{\beta^2}\|\nabla f(x_s)\|^2 \\
&\leq \|x_s - x*\|^2 - \frac{1}{\beta^2}\|\nabla f(x_s)\|^2 \\
&\leq \|x_s - x*\|^2
\end{aligned} \tag{19}$$

which concludes the proof.

4 Analyses of Stochastic Gradient Methods

In this section,We start by analyzing our SG algorithm when it is invoked to minimize a strongly convex objective function, where it is possible to establish a global rate of convergence to the optimal objective value [3]. To emphasize the generality of the results proved in this section, we remark that the objective function under consideration could be the expected risk or empirical risk ; i.e., we refer to the objective function $F : R^d \to R$, which represents $R(w) = E[f(w; \xi)]$ or $R_n(w) = \frac{1}{n}\sum_1^n f_i(w)$ This analyses apply equally to both objectives; the only difference lies in the way that one picks the stochastic gradient estimates in the method. The algorithm merely presumes that three computational tools exist: (i) a mechanism for generating a realization of a random variable ξ_k (with ξ_k representing a sequence of jointly independent random variables); (ii) given an iterate $w_k \in R^d$ and the realization of ξ_k , a mechanism for computing a stochastic vector $g(w_k, \xi_k) \in R^d$; and (iii) given an iteration number $k \in N$, a mechanism for computing a scalar stepsize $\alpha_k > 0$ [4] .

Algorithm 4.1 Stochastic Gradient Method

1: Choose an initial iterate w_1.
2: for $k = 1, 2, ...$ do
3: Generate a realization of the random variable ξ_k.
4: Compute a stochastic vector $g(w_k, \xi_k)$.
5: Choose a stepsize $\alpha_k > 0$.
6: Set the new iterate as $w_{k+1} \leftarrow w_k - \alpha_k g(w_k, \xi_k)$.
7: end for

Our analyses focus on two choices, one involving a fixed stepsize and one involving diminishing stepsizes, as both are interesting in theory and in practice. Notwithstanding all of this generality, we henceforth refer to Algorithm as SG.

4.1 Two Fundamental Lemmas

Our approach for establishing convergence guarantees for SG is built upon an assumption of smoothness of the objective function. This, and an assumption about the first and second moments of the stochastic vectors $g(w_k, \xi_k)$ lead to two fundamental lemmas from which all of our results will be derived. Our first assumption is formally stated as the following. Recall that, as already mentioned , F can stand for either expected or empirical risk.

Assumption 4.1 (Lipschitz-continuous objective gradients ,actually ,it's the same thing as (5)). The objective function $F : R^d \to R$ is continuously differentiable and the gradient function of F, namely, $\nabla F : R^d \to R$, is Lipschitz continuous with Lipschitz constant $L > 0$, i.e.,for all $w, \overline{w} \subset R^d$

$$\|\nabla F(w) - \nabla F(\overline{w})\|_2 \leq L\|w - \overline{w}\|_2 \tag{20}$$

Intuitively, Assumption 4.1 ensures that the gradient of F does not change arbitrarily quickly with respect to the parameter vector. Such an assumption is essential for convergence analyses of most gradient-based methods; without it,

the gradient would not provide a good indicator for how far to move to decrease F. An important consequence of Assumption 4.1 is that (actually,it's the same thing as lemma 3.1) for all $w, \overline{w} \subset R^d$

$$F(w) \leq F(\overline{w}) + \nabla F(\overline{w})^\top (w - \overline{w}) + \frac{1}{2} L \|w - \overline{w}\|_2^2 \tag{21}$$

Under Assumption 4.1 alone, we obtain the following lemma. In the result, we use $E_{\xi_k}[\cdot]$to denote an expected value taken with respect to the distribution of the random variable ξ_k . Therefore, $E_{\xi_k}[F(w_{k+1})]$ is a meaningful quantity since w_{k+1} depends on ξ_k through the update in Step 6 of Algorithm 4.1.
Lemma 4.1 Under Assumption 4.1, the iterates of SG (Algorithm 4.1) satisfy the following inequality for all $k \in N$:

$$E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq -\alpha_k \nabla F(w_k)^\top E_{\xi_k}[g(w_k, \xi_k)] + \frac{1}{2} \alpha_k^2 L E_{\xi_k}[\|g(w_k, \xi_k)\|_2^2] \tag{22}$$

Proof. By Assumption 4.1, the iterates generated by SG satisfy

$$F(w_{k+1}) - F(w_k) \leq \quad \nabla F(w_k)^\top (w_{k+1} - w_k) + \frac{1}{2} L \|w_{k+1} - w_k\|_2^2 \tag{23}$$

$$\leq \quad -\alpha_k \nabla F(w_k)^\top g(w_k, \xi_k) + \frac{1}{2} \alpha_k^2 L \|g(w_k, \xi_k)\|_2^2 \tag{24}$$

Taking expectations in these inequalities with respect to the distribution of ξ_k, and noting that w_{k+1} but not w_k depends on ξ_k, we obtain the desired bound. This lemma shows that, regardless of how SG arrived at w_k, the expected decrease in the objective function yielded by the kth step is bounded above by a quantity involving: (i) the expected directional derivative of F at w_k along $g(x_k, {}_k)$ and (ii) the second moment of $g(x_k, {}_k)$. For example, if $g(w_k, {}_k)$ is an unbiased estimate of $\nabla F(w_k)$, then it follows from Lemma 4.1 that

$$E_{\xi_k}[F(w(k+1))] - F(w_k) \leq -\alpha_k \|\nabla F(w_k)\|_2^2 + \frac{1}{2} \alpha_k^2 L E_{\xi_k}[\|g(w_k, \xi_k)\|_2^2] \tag{25}$$

We shall see that convergence of SG is guaranteed as long as the stochastic directions and stepsizes are chosen such that the right-hand side of (22) is bounded above by a deterministic quantity that asymptotically ensures sucient descent in F. One can ensure this in part by stating additional requirements on the first and second moments of the stochastic directions $g(w_k, \xi_k)$. In particular, in order to limit the harmful effect of the last term in (25), we restrict the variance of $g(w_k, \xi_k)$, i.e.

$$V_{\xi_k}[\|g(w_k, \xi_k)\|] = E_{\xi_k}[\|g(w_k, \xi_k)\|_2^2] + \|E_{\xi_k}[g(w_k, \xi_k)]\|_2^2 \tag{26}$$

Assumption 4.2 (First and second moment limits). The objective function and SG (Algorithm 4.1) satisfy the following:
(a) The sequence of iterates w_k is contained in an open set over which F is

bounded below by a scalar F_{inf}.
(b) There exist scalars $\mu_G \geq \mu > 0$ such that, for all $k \in N$,

$$\nabla F(w_k)^\top E_{\xi_k}[g(w_k, \xi_k)] \geq \mu \|\nabla F(w_k)\|_2^2 \quad (27)$$

$$\|E_{\xi_k}[g(w_k, \xi_k)]\|_2 \leq \mu_G \|\nabla F(w_k)\|_2 \quad (28)$$

(c) There exist scalars $M \geq 0$ and $M_v \geq 0$ such that, for all $k \in N$,

$$V_{\xi_k}[\|g(w_k, \xi_k)\|] \leq M + M_v \|\nabla F(w_k)\|_2^2 \quad (29)$$

The first condition, Assumption 4.2(a), merely requires the objective function to be bounded below over the region explored by the algorithm. The second requirement, Assumption 4.2(b), states that, in expectation, the vector $g(w_k, \xi_k)$ is a direction of sufficient descent for F from w_k with a norm comparable to the norm of the gradient. The properties in this requirement hold immediately with $\mu_G = \mu = 1$ if $g(w_k, \xi_k)$ is an unbiased estimate of $\nabla(w_k)$, and are maintained if such an unbiased estimate is multiplied by a positive denite matrix H_k whose eigenvalues lie in a fixed positive interval for all $k \in N$. The third requirement, Assumption 4.2(c), states that the variance of $g(w_k, \xi_k)$ is restricted, but in a relatively minor manner. For example, if F is a convex quadratic function, then the variance is allowed to be nonzero at any stationary point for F and is allowed to grow quadratically in any direction. All together, Assumption 4.2, combined with the definition (26), requires that the second moment of $g(w_k, \xi_k)$ satisfies

$$\|E_{\xi_k}[g(w_k, \xi_k)]\|_2^2 \leq M + M_G \|\nabla F(w_k)\|_2^2 \qquad M_G = M_v + \mu_G^2 \geq \mu^2 > 0 \quad (30)$$

Lemma 4.3. Under Assumptions 4.1 and 4.2, the iterates of SG (Algorithm 4.1) satisfy the following inequalities for all $k \in N$:

$$\begin{aligned} E_{\xi_k}[F(w(k+1))] \quad - \quad & F(w_k) \leq -\mu\alpha_k \|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha_k^2 L E_{\xi_k}[\|g(w_k, \xi_k)\|_2^2] \\ \leq \quad & -(\mu - \frac{1}{2}\alpha_k L M_G)\alpha_k \|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha_k^2 L M \end{aligned} \quad (31)$$

As mentioned, this lemma reveals that regardless of how the method arrived at the iterate w_k, the optimization process continues in a Markovian manner in the sense that w_{k+1} is a random variable that depends only on the iterate w_k, the seed ξ_k, and the stepsize α_k and not on any past iterates. This can be seen in the fact that the difference $E_k[F(w_{k+1})] - F(w_k)$ is bounded above by a deterministic quantity. Note also that the first term in (4.) is strictly negative for small ξ_k and suggests a decrease in the objective function by a magnitude proportional to $\|\nabla F(w_k)\|_2^2$. However, the second term in (31) could be large enough to allow the objective value to increase. Balancing these terms is critical in the design of SG methods.

4.2 SG for Strongly Convex Objectives

All of the convergence rate and complexity results presented in this section relate to the minimizaton of strongly convex functions. This is in contrast with

a large portion of the literature on optimization methods for machine learning, in which much effort is placed on attempts to strengthen convergence guarantees for methods applied to functions that are convex, but not strongly. My choice in this matter is based on a desire to present results that are more relevant to actual practice. After all, in much of machine learning practice, when a convex model is employed such as in logistic regression it is almost invariably regularized by a strongly convex function to facilitate the solution process. In particular, in my analyses, I focus on results that reveal the properties of SG iterates in expectation. The stochastic approximation literature, on the other hand, often relies on martingale techniques to establish almost sure convergence [5] [6] under the same assumptions [7]. For our purposes, I omit these complications since, in my view, they do not provide signicant additional insights into the forces driving convergence of the method. I formalize a strong convexity assumption as the following.

Assumption 4.3 (Strong convexity). The objective function $F : R^d \to R$ is strongly convex in that there exists a constant $c > 0$ such that for all $(\overline{w}, w) \in R^d \times R^d$

$$F(\overline{w}) \geq F(w) + \nabla F(w)^\top (\overline{w} - w) + \frac{1}{2} c \|\overline{w} - w\|_2^2 \tag{32}$$

Hence , F has a unique minimizer, denoted as $w_* \in R^d$ with $F_* := F(w_*)$. A useful fact from convex analysis is that, under Assumption 4.3, one can bound the optimality gap at a given point in terms of the squared ℓ_2-norm of the gradient of the objective at that point: for all $w \in R^d$

$$2c(F(w) - F_*) \leq \|\nabla F(w)\|_2^2 \tag{33}$$

We observe that, from (20) and (31), the constants in Assumptions 4.1 and 4.3 must satisfy $c \leq L$.

Theorem 4.1 (Strongly Convex Objective, Diminishing Stepsizes). Under Assumptions 4.1, 4.2, and 4.3 (with $F_{inf} = F_*$), suppose that the SG method (Algorithm 4.1) is run with a stepsize sequence such that, for all $k \in N$,

$\alpha_k = \frac{\beta}{\gamma + k}$ for some $\beta > \frac{1}{c\mu}$ and $\gamma > 0$ such that $\alpha_1 \leq \frac{\mu}{LM_G}$

Then, for all $k \in N$, the expected optimality gap satises

$$E[F(w_k) - F_*] \leq \frac{\nu}{\gamma + k} \tag{34}$$

where

$$\nu = max\Big(\frac{\beta^2 LM}{2(\beta c\mu - 1)}, (\gamma + 1)(F(w_1) - F_*)\Big) \tag{35}$$

Proof. the inequality $\alpha_k LM_G \leq \alpha_1 LM_G \leq \mu$ holds for all $k \in N$ Hence, along with Lemma 4.3 and (32), one has for all $k \in N$ that

$$\begin{aligned} E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq & \quad -(\mu - \frac{1}{2}\alpha_k LM_G)\alpha_k \|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha_k^2 LM \quad (36) \\ \leq & \quad -\frac{1}{2}\alpha_k \mu \|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha_k^2 LM \quad (37) \\ \leq & \quad -\alpha_k \mu c(F(w_k) - F_*) + \frac{1}{2}\alpha_k^2 LM \quad (38) \end{aligned}$$

Subtracting F_* from both sides, taking total expectations, and rearranging, this yields

$$E[F(w_{k+1}) - F_*] \leq (1 - \alpha_k \mu c)E[F(w_k) - F_*] + \frac{1}{2}\alpha_k^2 LM \quad (39)$$

We now prove (33) by induction. First, the definition of ν ensures that it holds for $k = 1$. Then, assuming (33) holds for some $k \geq 1$,(with $s = +k$) it follows from (38) that

$$\begin{aligned} E[F(w_{k+1}) - F_*] \leq & \quad \big(1 - \frac{\beta\mu c}{s}\big)\frac{\nu}{s} + \frac{\beta^2 LM}{2s^2} \\ = & \quad \big(\frac{s - \beta\mu c}{s^2}\big)\nu + \frac{\beta^2 LM}{2s^2} \\ = & \quad \big(\frac{s-1}{s^2}\big)\nu - \big(\frac{\beta\mu c - 1}{s^2}\big)\nu + \frac{\beta^2 LM}{2s^2} \\ \leq & \quad \frac{\nu}{s+1} \quad (40) \end{aligned}$$

where the last inequality follows because $s^2 \geq (s+1)(s-1)$

Theorem 4.2 (Strongly Convex Objective, Fixed Stepsize)
Under Assumptions 4.1, 4.2, and 4.3 (with Finf = F), suppose that the SG method (Algorithm 4.1) is run with a fixed stepsize, $_k =$ for all $k \in N$, satisfying

$$0 < \alpha \leq \frac{\mu}{LM_G} \quad (41)$$

Then, the expected optimality gap satisfies the following inequality for all $k \in N$

$$E[F(w_{k+1}) - F_*] \leq \frac{\alpha LM}{2c\mu} + (1 - \alpha c\mu)^{k-1}\Big(F(w_1) - F_* - \frac{\alpha LM}{2c\mu}\Big) \xrightarrow{k\to\infty} \frac{\alpha LM}{2c\mu} \quad (42)$$

Proof. Using Lemma 4.3 with (33) and (41), we have for all $k \in N$ that

$$\begin{aligned} E_{\xi_k}[F(w_{k+1})] - F(w_k) \leq & \quad -(\mu - \frac{1}{2}\alpha LM_G)\alpha \|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha^2 LM \\ \leq & \quad -\frac{1}{2}\alpha\mu \|\nabla F(w_k)\|_2^2 + \frac{1}{2}\alpha^2 LM \\ \leq & \quad -\alpha_k \mu c(F(w_k) - F_*) + \frac{1}{2}\alpha^2 LM \quad (43) \end{aligned}$$

Subtracting F_* from both sides, taking total expectations, and rearranging, this yields

$$E[F(w_{k+1}) - F_*] \leq (1 - \alpha\mu c)E[F(w_k) - F_*] + \frac{1}{2}\alpha^2 LM \tag{44}$$

Subtracting the constant $\alpha LM/(2c\mu)$ from both sides, one obtains

$$\begin{aligned} E[F(w_{k+1}) - F_*] - \frac{\alpha LM}{2c\mu} \leq & \quad (1 - \alpha\mu c)E[F(w_k) - F_*] + \frac{1}{2}\alpha^2 LM - \frac{\alpha LM}{2c\mu} \\ \leq & \quad (1 - \alpha c\mu)\big(E[F(w_k) - F_*] - \frac{\alpha LM}{2c\mu}\big) \end{aligned} \tag{45}$$

Observe that (45) is a contraction inequality since, by (41) and (30),

$$0 < \alpha \leq \frac{c\mu^2}{LM_G} \leq \frac{c\mu^2}{L\mu^2} = \frac{c}{L} \leq 1 \tag{46}$$

The result thus follows by applying (45) repeatedly through iteration $k \in N$. Theorem 4.2 illustrates the interplay between the stepsizes and bound on the variance of the stochastic directions. If there were no noise in the gradient computation or if noise were to decay with $\|\nabla F(w_k)\|_2^2$(i.e., if $M = 0$ in (29) and (30)), then one can obtain linear convergence to the optimal value. This is a standard result for the full gradient method with a sufficiently small positive stepsize. On the other hand, when the gradient computation is noisy, one clearly loses this property. One can still use a xed stepsize and be sure that the expected objective values will converge linearly to a neighborhood of the optimal value, but, after some point, the noise in the gradient estimates prevent further progress; It is apparent from (42) that selecting a smaller stepsize worsens the contraction constant in the convergence rate, but allows one to arrive closer to the optimal value.

Let us now remark on what can be learned from Theorems 4.1 and 4.2.

4.3 Role of Strong Convexity

Observe the crucial role played by the strong convexity parameter $c > 0$, the positivity of which is needed to argue that (45) and (39) contract the expected optimality gap. However, the strong convexity constant impacts the stepsizes in dierent ways in Theorems 4.1 and 4.2. In the case of constant stepsizes, the possible values of α are constrained by the upper bound (41) that does not depend on c. In the case of diminishing stepsizes, the initial stepsize α_1 is subject to the same upper bound , but the stepsize parameter β must be larger than $\frac{1}{c\mu}$. This additional requirement is critical to ensure the $O(\frac{1}{k})$ convergence rate. How critical? Consider, e.g., [8] in which the authors provide a simple example (assuming $\mu = 1$) involving the minimization of a deterministic quadratic

function with only one optimization variable in which c is overestimated, which results in β being underestimated. In the example, even after 10^9 iterations, the distance to the solution remains greater than 10^{-2}.

4.4 Role of the Initial Point

Also observe the role played by the initial point, which determines the initial optimality gap, namely, $F(w_1) - F_*$. When using a fixed stepsize, the initial gap appears with an exponentially decreasing factor; see (42). In the case of diminishing stepsizes, the gap appears prominently in the second term defining ν in (35). However, with an appropriate initialization phase, one can easily diminish the role played by this term. For example, suppose that one begins by running SG with a fixed stepsize α until one (approximately) obtains a point, call it w_1, with $F(w_1) - F \leq \frac{\alpha LM}{2c\mu}$. A guarantee for this bound can be argued from (42). Starting here with $^1 =$, the choices for β and γ in Theorem 4.1 yield

$$(\gamma + 1)E[F(w_1) - F_*] \leq \frac{\beta}{\alpha_1}\frac{\alpha LM}{2c\mu} = \frac{\beta LM}{2c\mu} < \frac{\beta^2 LM}{2(\beta c\mu - 1)} \tag{47}$$

meaning that the value for ν is dominated by the first term in (35).
On a related note, we claim that for practical purposes the initial stepsize should be chosen as large as allowed, i.e., $\alpha_1 = \mu/(LM_G)$. Given this choice of α_1, the best asymptotic regime with decreasing stepsizes (34) is achieved by making ν as small as possible. Since we have argued that only the first term matters in the definition of ν, this leads to choosing $\beta = 2/(c\mu)$. Under these conditions, one has

$$\nu = \frac{\beta^2 LM}{2(\beta c\mu - 1)} = \frac{2}{\mu^2}\left(\frac{L}{c}\right)\left(\frac{M}{c}\right) \tag{48}$$

We shall see the (potentially large) ratios L/c and M/c arise again later
Trade-Offs of (Mini-)Batching As a final observation about what can be learned from Theorems 4.1 and 4.2, let us take a moment to compare the theoretical performance of two fundamental algorithmsthe simple SG iteration and the mini-batch SG iteration when these results are applied for minimizing empirical risk, i.e., when $F = R_n$. This provides a glimpse into how such results can be used to compare algorithms in terms of their computational trade-offs. The most elementary instance of our SG algorithm is simple SG, which, as we have seen, consists of picking a random sample index i_k at each iteration and computing

$$g(w_k, \xi_k) = \nabla f_{i_k}(w_k) \tag{49}$$

By contrast, instead of picking a single sample, mini-batch SG consists of randomly selecting a subset S_k of the sample indices and computing

$$g(w_k, \xi_k) = \frac{1}{|S_k|}\sum_{i \in S_k} \nabla f_{i_k}(w_k) \tag{50}$$

To compare these methods, let us assume for simplicity that we employ the same number of samples in each iteration so that the mini-batches are of constant size, i.e.,$|S_k| = n_{mb}$. There are then two distinct regimes to consider, namely, when $n_{mb} \ll n$ and when $n_{mb} \approx n$. Our goal here is to use the results of Theorems 4.1 and 4.2 to show that, in the former scenario, the theoretical benet of mini-batching can appear to be somewhat ambiguous, meaning that one must leverage certain computational tools to benefit from mini-batching in practice. As for the scenario when $n_{mb} \approx n$, the comparison is more complex due to a trade-off between per-iteration costs and overall convergence rate of the method . We leave a more formal treatment of this scenario, specically with the goals of large-scale machine learning in mind.
Suppose that the mini-batch size is $n_{mb} \ll n$. The computation of the stochastic direction $g(w_k, \xi_k)$ in (50) is clearly n_{mb} times more expensive than in (49). In return, the variance of the direction is reduced by a factor of $\frac{1}{n_{mb}}$. That is, with respect to our analysis, the constants M and $M_V$that appear in Assumption 4.2 are reduced by the same factor, becoming M/n_{mb} and M_V/n_{mb} for mini-batch SG. It is natural to ask whether this reduction of the variance pays for the higher per-iteration cost. Consider, for instance, the case of employing a sufficiently small stepsize $\alpha > 0$. For mini-batch SG, Theorem 4.2 leads to

$$E[F(w_k) - F_*] \leq \frac{\alpha LM}{2c\mu n_{mb}} + [1 - \alpha c\mu]^{k-1}\Big(F(w_1) - F_* - \frac{\alpha LM}{2c\mu n_{mb}}\Big) \qquad (51)$$

Using the simple SG method with stepsize α/n_{mb} leads to a similar asymptotic gap:

$$E[F(w_k) - F_*] \leq \frac{\alpha LM}{2c\mu n_{mb}} + [1 - \frac{\alpha c\mu}{n_{mb}}]^{k-1}\Big(F(w_1) - F_* - \frac{\alpha LM}{2c\mu n_{mb}}\Big) \qquad (52)$$

The worse contraction constant (indicated using square brackets) means that one needs to run n_{mb} times more iterations of the simple SG algorithm to obtain an equivalent optimality gap. That said, since the computation in a simple SG iteration is n_{mb} times cheaper, this amounts to effectively the same total computation as for the mini-batch SG method. A similar analysis employing the result of Theorem 4.1 can be performed when decreasing stepsizes are used. These observations suggest that the methods can be comparable. However, an important consideration remains. In particular, the convergence theorems require that the initial stepsize be smaller than $\mu/(LM_G)$. Since (30) shows that$M_G \geq \mu^2$, the largest this stepsize can be is $1/(\mu L)$. Therefore, one cannot simply assume that the mini-batch SG method is allowed to employ a stepsize that is n_{mb} times larger than the one used by SG. In other words, one cannot always compensate for the higher per-iteration cost of a mini-batch SG method by selecting a larger stepsize. One can, however, realize benefits of mini-batching in practice since it offers important opportunities for software optimization and parallelization; e.g., using sizeable mini-batches is often the only way to fully leverage a GPU processor. Dynamic mini-batch sizes can also be used as a substitute for decreasing stepsizes.

5 Summary

Actually, this topic still has a lot of things to do. e.g., we can consider non-smooth fuction. And also we can pay attention to other GD methods,such as Momentum [9],Nesterov accelerated gradient [10] and Adagrad [11]. They are all interesting. However, take time and energy into consideration,I just focus on two parts, which I think is most important and meaningful. At first, I provide a proof of convergence theory for GD in the unconstrained situation, where f is a convex and β-smooth function on R^n. Then a concise, yet broadly applicable convergence and complexity theory for SG is presented here, providing insight into how these guarantees have translated into practical gains.

References

[1] Stephen Boyd and Lieven Vandenberghe. Convex Optimization. Cambridge UP, 2004. Online: http://www.stanford.edu/ boyd/cvxbook/

[2] Bahlak S, Gazalet J, Lefebvre J E, et al. Convex Optimization: Algorithms and Complexity[J]. Foundations Trends in Machine Learning, 2014, 8(3-4):231-357.

[3] Bottou L, Curtis F E, Nocedal J. Optimization Methods for Large-Scale Machine Learning[J]. 2016.

[4] Robbins H, Monro S. A Stochastic Approximation Method[J]. Annals of Mathematical Statistics, 1951, 22(3):400-407.

[5] E. G. Gladyshev, On stochastic approximations, Theory of Probability and its Applications, 10 (1965), pp. 275278.

[6] H. Robbins and D. Siegmund, A convergence theorem for non negative almost supermartingales and some applications, in Optimizing Methods in Statistics, J. S. Rustagi, ed., Academic Press, 1971.

[7] L. Bottou, Online Algorithms and Stochastic Approximations, in Online Learning and Neural Networks, D. Saad, ed., Cambridge University Press, Cambridge, UK, 1998.

[8] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, Robust Stochastic Approximation Approach to Stochastic Programming, SIAM Journal on Optimization, 19 (2009), pp. 1574 1609.

[9] Ning Qian. On the momentum term in gradient descent learning algorithms. Neural networks : the ofcial journal of the International Neural Network Society, 12(1):145151, 1999.

[10] Su W, Boyd S, Candes E J. A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights[J]. Advances in Neural Information Processing Systems, 2015, 3(1):2510-2518.

[11] Diederik P. Kingma and Jimmy Lei Ba. Adam: a Method for Stochastic Optimization. International Conference on Learning Representations, pages 113, 2015.